# Evaluation of Yelp User Review Effectiveness

**Yuan Liang**
yl4ps@virginia.edu

**Dong Xu**
dx3yy@virginia.edu

**Muyun Lu**
ml4ra@virginia.edu

## 1 Introduction

Our project is to use Yelp dataset to extract full review text from each $user_id$ in review file, which contains several sentances. We use natural language pre processing and transfer text into 5 levels, corresponding to the star feature. After that we use K means clustering algorithms, combain the NLP output with stars, useful, cool and funny to evaluate the Yelp rating system.

The motivation of this problem is to study if the "5-star" review system of Yelp could effectively and accurately reflect how the Yelp users review the business or this mechanism is totally subjective which doesn't match user reviews. The definition to this study is data clustering, which is to find natural groupings in our text data, is an interesting aspect of machine learning and pattern recognition.

Finally we get some conclusions. It's not easy task to design a model to perform sentiment analysis of the user reviews of the Yelp dataset. K-means is able to group components of user review into four sensical clusters. There are some feature redundancy in the design of Yelp dataset.

## 2 Natural Language Preprocessing

For Natural Language Preprocessing, we use Tokenization, Normalization and Vectorization.

Given a character sequence and a defined document unit, tokenization is the task of chopping it up into pieces, called tokens, perhaps at the same time throwing away certain characters, such as punctuation.

Token normalization is the process of canonicalizing tokens so that matches occur despite superficial differences in the character sequences of the tokens. The most standard way to normalize is to implicitly create equivalence classes, which are normally named after one member of the set.

Vectorization is basic model used in natural language processing. It is called bag of words because any order of the words in the document is discarded. It only tells us weather word is present in the document or not.

## 3 TF-IDF

In vectorization, we use TF-IDF. TF-IDF stands for Term Frequency-Inverse Document Frequency which basically tells importance of the word in the corpus or dataset. TF-IDF contain two concept Term Frequency(TF) and Inverse Document Frequency(IDF).

Term Frequency is defined as how frequently the word appear in the document or corpus. As each sentence is not the same length so it may be possible a word appears in long sentence occur more time as compared to word appear in sorter sentence. Term frequency can be defined as: $TF = \frac{Frequency\,of\,word}{Total\,number\,of\,word}$

Preprint. Under review.

Inverse Document frequency is another concept which is used for finding out importance of the word. It is based on the fact that less frequent words are more informative and important. IDF is represented by formula: $IDF = \log \frac{Number\,of\,document}{Number\,of\,document\,the\,word\,exists}$

TF-IDF is basically a multiplication between TF table and IDF table . It basically reduces values of common word that are used in different document.

## 4 Proposed Idea

This is an application project where the plan mainly contains 3 stages: text review pre-processing, sentimental analysis by NLP, where we have tried different models, and unsupervised learning via K-means method clustering to evaluate the Yelp rating system. The pipeline we built for NLP during the first stage contains three different parts: tokenization, normalization and vectorization. We used nltk for these tasks. Tokenization is to remove punctuation by replacing white spaces as well as split text into tokens, then drop those most commonly appeared but not sentiment related words as 'stopwords'. Normalization is to transfer those tokens into more uniform format, such as to make all words lower case, to remove suffixes or prefixes and lemmatization, which means to convert verb to its basic form via morphological analysis. We use tf-idf algorithm which is a numerical statistic method to calculate the weight of each word. The term frequency – inverse document frequency means that, value increases proportionally to the number of times a word appears in the document and is offset by the number of documents in the dataset that contain the word. The weighting scheme for this algorithm looks like this:

$$(1 + log f_{t,d}) log(1 + \frac{N}{n_t})$$

The main algorithm we will use for the unsupervised learning is K means algorithm. This iterative clustering method follows expectation-maximization by assigning the data points to the closest cluster while computing the centroid of each cluster. If time permits we would also like to test several variations of this algorithm including K-harmonic means or Fuzzy k-means, to experiment and study if there is an improvement or not.

The last stage is to evaluate the outcome of this project. If we could successfully convert the text review data into normalized multidimensional vectors via pre-processing, we would like to see that there might be around 5 separated groups of text review data point after the clustering, which matches the "5-star" evaluation mechanism used on Yelp website. We could also use silhouette analysis to determine the degree of separation between these groups by computing the average distance from all data points in the same cluster $a^i$ or the closest cluster $b^i$, then check the silhouette value:

$$s(i) = \frac{b^i - a^i}{max(a^i, b^i)}$$

## 5 Related Work

Thanks to the resourceful information provided in the Yelp dataset, researchers have utilized it to result great works. [3, 4, 5] Among them, stands out an interesting work which built a recommender system based on personalizing various diversity tolerances which are estimated by using clustering techniques. [3] To be particular, they proposed a way to personalize diversity by performing collaborative filtering independently on different classes of users based on their degree of diversity; they also investigated the accuracy-diversity trade-offs on different classes of users with novel metrics and shown the effectiveness of their approach.

This paper [2] investigate potential factors that may affect business performance on Yelp. It used a mix of features already available in the Yelp dataset and generating own features using location clustering and sentiment analysis of reviews. After preprocessing the data to handle missing values, muti-class classification was run on the feature subsets with an accuracy of 45%, significantly higher than random chance of 16.7%. Regression models were also tested but achieved lower accuracy.

In the paper <Comparison of Clustering Methods using YELP dataset>[1], they experimented with several common clustering methods including: K-means, Bisecting K-means, Spectral Clustering,

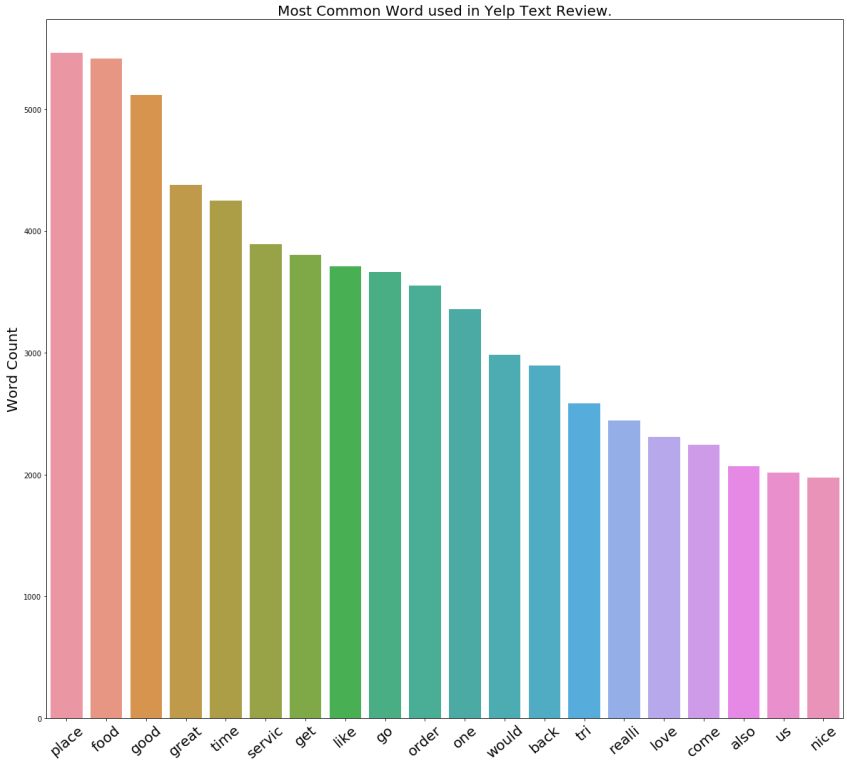
Gaussian Mixture Model and Power Iteration Clustering(PIC). The study is done on Yelp dataset. From analyze the results of this study, one may conclude that the GMM method assumes the dataset has a Gaussian distributions which is not the case, meanwhile, PIC needs a dominant eigenvalue which doesn't exist in the Yelp dataset. The Bisecting K-means and K-means method both give similar but still quite low performance at a 0.58 silhouette value, which might be caused by the Yelp dataset not being spherical as usually required by K-means method. But the K means method still has the best performance overall on this specific dataset.

# 6 Experiment

## 6.1 Experiment Setup

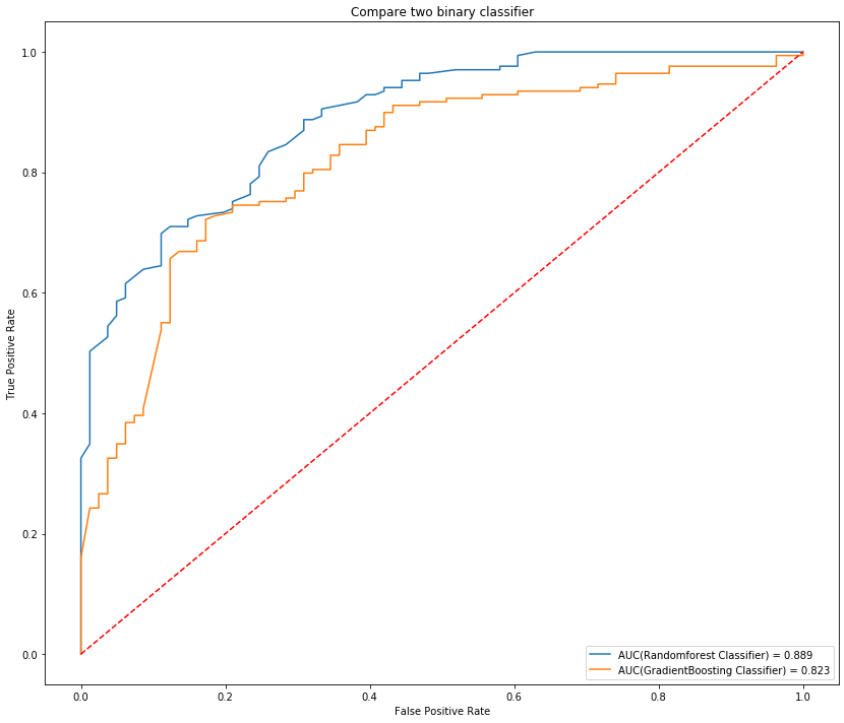
The setup of experiment is to use the first 10000 data from 'review.json' file as training dataset for NLP process and the following 2000 data for validation purpose. As a direct result of normalization process, we can do a simple check on the most frequently appeared words in the text reviews, as you can see in Figure[1], most of the reviewers seem to have positive opinions because words like "good,

Figure 1: Most common words used in Yelp review



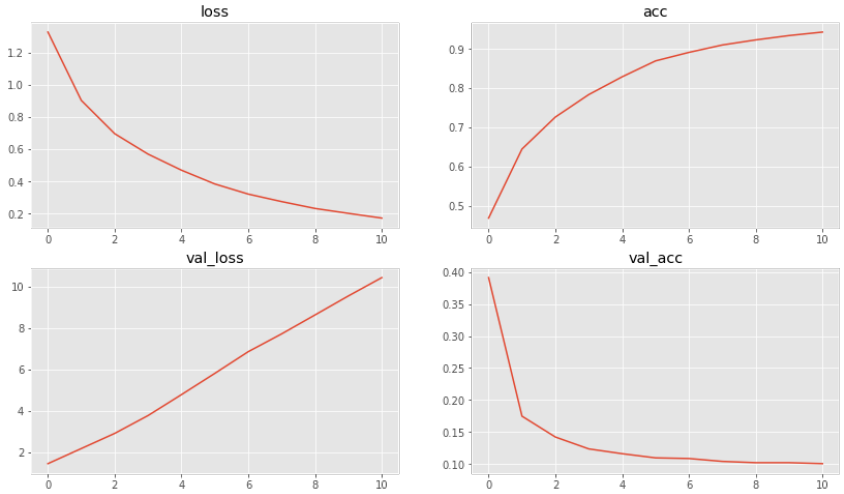

great, like, love and nice" are more frequent.
For the training process, we firstly tried with binary output that only includes 'positive' or 'negative' by using random forest and gradient boosting classifier. Random forest is an ensemble learning method that operate by constructing a multitude of decision tress at training time and outputting the class that is the mode of classes. Gradient boosting is also an ensemble learning method by allowing optimization of an arbitrary differentialable loss function. We compared these method by calculating its metrics, that is accuracy, precision and recall. The accuracy for the random forest method is 0.796 which is slightly better than gradient boosting with 0.776. As a further comparison, we plotted the AUC curve in Figure[2]. It seems that random forest has overall better performance than gradient boosting method.
However, Yelp is using a 5-star rating system, if we would like to directly match our sentimental results from text review to this, the output should be categorical with size of 5. In order to achieve this, after pre-processing we firstly did a dimensionality reduction to limit the size of word bag. This

Figure 2: AUC curve

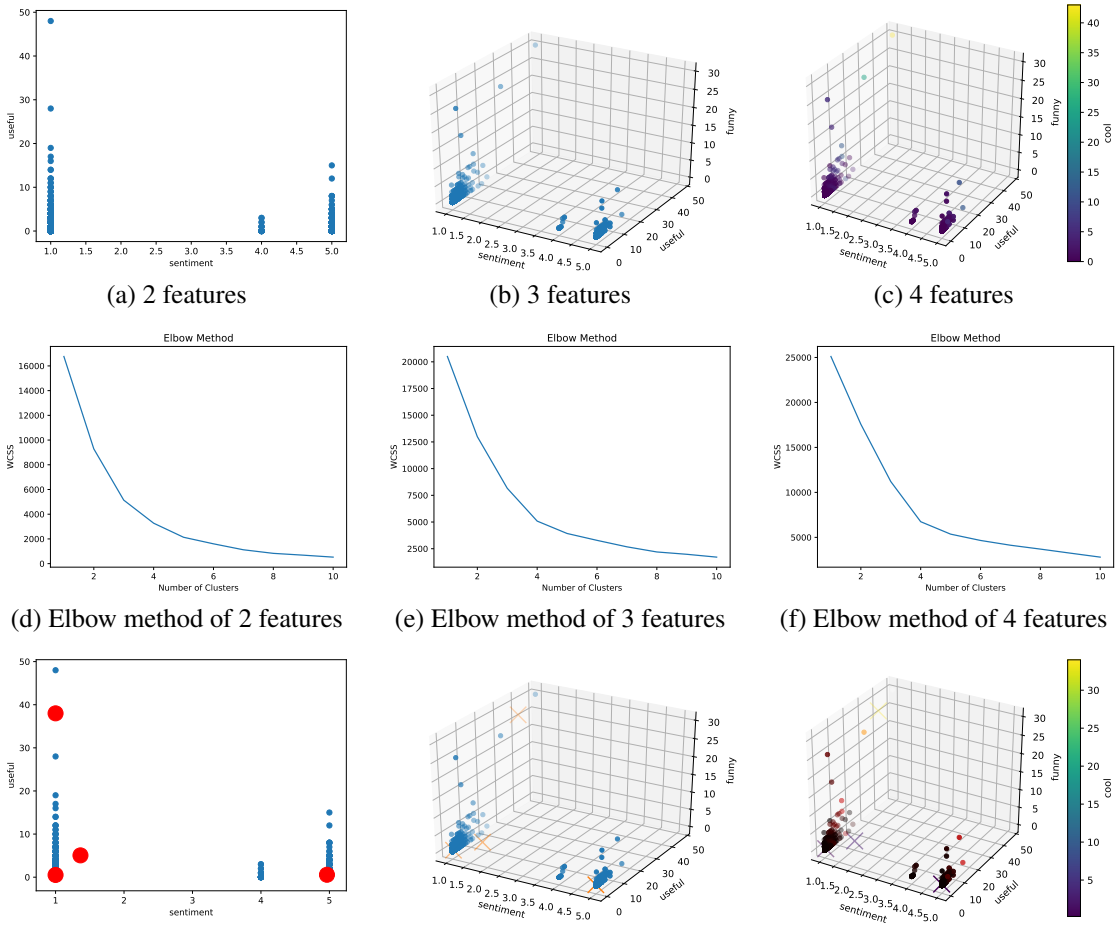



value is also used as a hyper-parameter in our training model. Then we decide to use a multilayer perceptron as our classifier where we use Keras package to change the binary output into categorical ones. Our model has an input layer, a hidden layer and an output layer, which the hidden layer has a dropout rate of 0.3. The learning rate is $10^{-3}$. However from the metrics here in Figure[3], although the training model reaches above 90% accuracy within 10 epochs, the validation loss and

Figure 3: Metrics of MLP model for NLP




validation accuracy looks terrible. Also the results of predicted sentimental value is tilted compared with the actual value. One possible reason could be that, although it might be easier to tell very positive review from very negative review, for the neutral reviews that has either 2-star or 4-star, it's ambiguous from 1-star or 5-star given the fact those review texts are subjective and the words used totally depends on the reviewer's habit. So that it might be rather difficult to find a robust model for




(a) 2 features (b) 3 features (c) 4 features

(d) Elbow method of 2 features (e) Elbow method of 3 features (f) Elbow method of 4 features

(g) K-means of 2 features (h) K-means of 3 features (i) K-means of 4 features

Figure 4: K-means results

this specific sentiment analysis.

### 6.2 Experiment Results

Here, we analyzed the k-means clustering results and made three types of plots. In this case, we used 2000 data points and four feature. The features are combinations of the results of the sentiment predictions using our NLP model over the user review texts, of useful, of funny, and useful clicks this review received. We used the elbow method to determine the number of clusters.

In figure 4(a,b,c), we show the distribution of the data points of using 2,3,and 4 features respectively. We can see that the data are classed in about 3 groups for all three combinations. We then used the Elbow method as shown in figure 4(d,e,f). The 3 combinations show similar trend. Based on the curvature of the plot, we chose 4 to be the number of clusters to use for k-means. We then calculated the k-means of each data collections and plotted the center of the 4 clusters on top of the data distribution plots as shown in figure 4(g,h,i). Again, the centers of the clusters are almost at the same position for the 3 different feature combinations. We can see that, k-means clusters the observations near 5 sentiment values into 1 cluster because there wasn't much variation on the other axises. It also grouped other observations near 1 sentiment values into 3 clusters as the observations are more sparse here.

### 6.3 Experiment Analysis

Based on our plotted results, we found some interesting trends. Firstly, the sentiment values are most negative at 0 with some positive at 4 and 5 ranks. There are no ranks of 2 and 3. We believe that this could be caused by the incompetency of the NLP model to differentiate the medium attitude of the review texts because when during training the data is not well distributed. Also, we found kmeans is able to cluster the data into 4 group which was close to what we expected. We originally though there should be only 3 cluster. However, this could happen as the data is very sparse at the lower left corner as shown in the figure. Lastly, for 3 different feature combinations, the data distribution, the elbow method, and the k-means results are very similar. We conjecture that it is because the useful, funny, and cool features has a very positive correlations with each other that diminished the nature of each feature.

## 7 Conclusion

To conclude, we found it is not a easy task to train a model to perform sentiment analysis of the user text review of the Yelp dataset due to their complexity. We were able to use k-means to group user review features into 4 sensical clusters. We also found there are some redundant features in the design of the Yelp dataset.

To improve our project, there are three main directions. The first is to train a better NLP model to perform sentiment analysis. The second is to try more features, more data points, other features, and other objects, etc. And lastly, we can more precise methods to determine the number of clusters to use.

## References


[1] A. O. Angadpreet Nagpal. Comparison of clustering methods using yelp dataset. 2018.

[2] C. Cyle, K. Fujii, and P. Veerina. *Application to machine learning to predict Yelp ratings*. 2014.

[3] F. Eskandanian, B. Mobasher, and R. Burke. A clustering approach for personalizing diversity in collaborative recommender systems. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*, pages 280–284, 2017.

[4] J. Huang, S. Rogers, and E. Joo. Improving restaurants by extracting subtopics from yelp reviews. *iConference 2014 (Social Media Expo)*, 2014.

[5] T. Jindal. *Finding local experts from Yelp dataset*. PhD thesis, 2015.